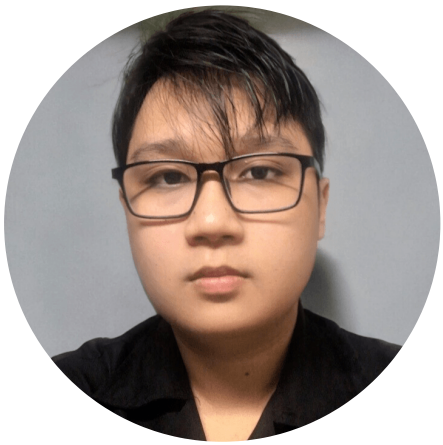
Tran Dong Tri

DevOps Fresher



With nearly a year of experience working with Back-end **.NET** through researching and implementing personal projects surrounding website technology, mobile applications and Desktop applications. Through that, I learned more about knowledge about **Linux** operating systems, **AWS**, **Terraform**, **Docker**, **Gitlab**, and **Jenkins**. I hope to be able to apply the knowledge I have learned in a professional working environment. Become a professional DevOps engineer, bringing value to the company.

Ho Chi Minh city 0363606843 http://tylertran.click

Education

BTEC FPT INTERNATIONAL COLLEGE

College degree in software engineering

GPA: 8.62/10

2021 – 2023.

NHA TRANG COLLEGE OF TECHNOLOGY

Level 4 of Vietnam Qualification Framework

Graduation rating: Good

2019 – 2021.

Achievement

- Excellent student in: Summer 2021, Spring 2022, Fall 2022, Summer 2023.
- Excellent student in English learning phase.

Language

Confidently communicate in English.

Certificate

- Google cloud training day(CORE INFRASTRUCTRURE FUNDAMETAIL)
- DevOps for Freshers by Devopsedu

Technical skills

DevOps skills:

- **CICD tools: Jenkins**, **Gitlab CI**
- **Container**: **Docker,** Kubernetes (basic)
- **Database**: MySQL, MariaDB.
- **Monitor:** Zabbix
- **webserver:** Nginx, Apache 2
- **IaC (basic):** Terraform
- **AWS services (basic):** IAM, EC2, VPC, S3, Cloud9, ALB ,ECS, ECR, CloudMap, CodeCommit
- **Linux system**, **Bash command**

Developer Skill:

- **Language**: **C#**, **XAML**, Dart
- **Technology**: **WPF**, **ASP.NET**, **.NET Core API, Winform**, Flutter, Symfony.
- **Database**: **SQL Server**, SQLite, PostgreSQL.
- **Tools**: Github, **Gitlab**, Visual Studio, Visual studio Code, Postman

Experience

LNT Nha Trang (06/2021 - 08/2021)

Position: Tester intern.

Responsibility:

- Write test cases for restaurant management application.
- Draw a risk management chart from detected errors.
- Test and detect errors related to: function, performance, data input errors.
- Participate in and develop restaurant management applications:
  - Table reservation function and interface with xaml.
  - Ordering function interface.
  - Employee management function interface.

Devops Projects

Jenkins CI/CD pipeline .NET and ReactJS projects in a staging environment

Personal project | 12/03/2024

- Tools: **Jenkins**, **Gitlab, Sonarqube, Zabbix, Docker, Slack App**
- Technology: .**Net (backend), ReactJs (Frontend)**
- Detail:
  - Connection and configuration Slack chanel to get notifycation from Jenkins CI/CD
  - Configuration and settings for 4 servers include:
    - Server1 (Staging deploy server)
    - Server2 (Gitlab server)
    - Server3 (Jenkins server)
    - Server5 (Sonarqube server)
    - Server4 (Zabbix monintoring server)
  - Describe:
    - The developer will push the code to the private Gitlab server, then Jenkins will receive the trigger and execute the CI/CD pipeline.
    - In CI/CD Jenkins will use the Sonarqube server to test the code quality of the .NET API. The pipeline will then have database migration operations.
    - Finally, Jenkins will upload those images to Docker Hub and connect to the deployment server using SSH to pull the image back and start deploying the website.
    - Jenkins will report success or errors from the link to the linked Slack channel.
  - Implement the CI/CD Pipeline process with CD as Continuous Deployment in theses stages:
    - Check source
    - Test with dotnet
    - Test with sonarqube
    - Migration database
    - Build and push images
    - Deploy to staging

Implementation process: Jenkins CI/CD pipeline .NET and ReactJS projects in a staging environment

Jenkins CI/CD(CD:Continuous Deliver) spring boot java project in production enviroment

Personal project | 02/03/2024

- Tools: **Jenkins**, **Gitlab**, **Zabbix**
- Technology: Java Spring Boot
- Detail:
  - Configuration and settings for 4 servers include:
    - Server1 (Jenkins node, Java, Maven, Tomcat)
    - Server2 (Gitlab server)
    - Server3 (Jenkins server)
    - Server4 (Zabbix monintoring server)
  - Connect the Jenkins server to the Gitlab server to be able to pull the source code.
  - Connect the Jenkins server to the Jenkins node server to run the project.
  - Connect the Zabbix server to the remaining 3 servers (Jenkins, Jenkins node, Gitlab) to monitor and signal alarms of the servers.
  - Implement the CI/CD Pipeline process with CD as Continuous Delivery and parameters. Includes actions:
    - Start: Launch the project (stages: start)
    - Stop: Stop the project (stages: stop)
    - Update: Update new project functionality using commit ID from Gitlab. (stages: backup, stop, checkout, build, config, start)
    - Rollback: Relaunch the project in the previous version. (stages: stop, rollback, start)

Implementation process: Jenkins CI/CD(CD:Continuous Deliver) spring boot java project in production enviroment

Create CodePipeline - Deploy to ECS using Blue-Green

Personal project | 11/04/2024

- AWS Services: CodePipline, CodeCommit, CodeBuild, ECR, ECS, ALB
- Technology: NodeJs
- Detail:

- Create a CI/CD pipeline including stages: Source, Build, Deploy.
  - Source: Developer will push the code into the repository created by CodeCommit.
  - Build: After receiving the event, CodeCommit will trigger CodeBuild to build a Docker image and then push that image to the repository created by ECR. At the same time, the configuration will be saved to S3 Bucket.
  - Deploy: Create an application from CodeDeploy with blue-green deployment mode. The controller creates 2 tasks for the blue item and 2 tasks for the green item. These 4 EC2s will be distributed load-balanced by ALB. These EC2 devices will be pulled images and deployed with 2 ports 80 and 81. Blue deployment is for end users and Green deployment is for application testing. After testing is complete, the EC2s in the blue deployment section will clone the new version from the green deployment section, and the EC2s in the green deployment section will be deleted.

Implementation process: **Create CodePipeline - Deploy to ECS using Blue-Green**

Create a job deploy to ElaticBeanstalk (pet project)

Personal project | 10/04/2024

- AWS Services: CodePipline, CodeCommit, CodeBuild, Elastic Beanstalk, S3
- Technology: Java
- Detail:
  - Create a CI/CD pipeline including stages: Source, Build, Deploy.
    - Source: Developer will push the code into the repository created by CodeCommit.
    - Build: After receiving the event, CodeCommit will trigger CodeBuild, which will build an artifact of that technology. The newly created artifact will then be saved to the corresponding S3 Bucket.
    - Deploy: After the project build event is successful, Codepipeline will direct Elastic Beanstalk to retrieve the newly saved artifact and deploy it to a pre-configured EC2.

Implementation process: **Create a job deploy to ElaticBeanstalk**

Setup Gitlab CI/CD pipeline with private registry (pet project)

Personal project | 03/202

- Tools: **Docker**, **Gitlab**, Harbor Registry, OpenSSL, AWS EC2, Cloudflare domain manager
- Technology: Java Spring Boot
- Detail:
  - Configuration and settings for 3 servers include:
    - Server1 (Gitlab runner, Java, Maven, Tomcat)
    - Server2 (Gitlab server)
    - AWS EC2 (Regisity with Harbor)
  - Install Gitlab and deploy the project to Gitlab in server2
  - Install Gitlab-runner in server1 to connect to Gitlab server
  - Implement the CICD pipeline configuration on Gitlab and in the project source
  - Set up Registry in AWS EC2 connection with deploy server (server1)
  - Use docker to package projects, push images and pull images from Registry (AWS EC2)
  - CI/CD pipeline stages:
    - build
    - deploy

Developer Projects

Dating App

Personal project | 06/2023 - 08/2023

- Programming language: **C#**, TypeScript
- Technology: MVC, **.NET Core**, AngularJs, SQLite, PostgresSQL, Bootstrap, Fly.io
- Tools: Microsoft Visual Code, Postgres, Docker
  - Build separate API (.NET Core) and Client (AngularJS) systems.
  - Develop a system to list dating partners through filters (filter by gender, age), and can follow dating partner.
  - Develop realtime messaging system using SignalR.
  - Deploy applications using Fly.io cloud service.
  - Save image files using the Cloudinary cloud storage service

Github: TylerTran4201/DatingApp.git

Restaurant Management

Personal project | 09/2022 - 01/2023

- Programming language: **C#**

- Technology: MVVM Model, **.NET WPF**, SQLServer
- Tools: Microsoft Visual Studio
  - Develop an employee interaction system that implements the following features: Table reservation, table separation, ordering, payment, and bill payment.
  - The Admin side can manage information and perform basic CRUD operations in SQL Server.
- Github: TylerTran4201/QuanLyNhaHang_WPF

Library Management

Personal project | 08/2022 - 09/2022

- Programming language: **C#**
- Technology: MVVM Model, **.NET WPF**,
- Tools: Microsoft Visual Code.
  - Develop a library management system for library managers. Includes the following features: reader management, book management, category management, book lending, and book return (functions for readers).
  - Data is saved in XML files.
- Github: TylerTran4201/LibraryManagement_WPF

Greendit

Personal project | 09/2023 - 12/2023

- Programming language: Dart
- Technology: Flutter, Firebase, TensorFlow
- Tools: Android Studio
  - Develop a simple social network that includes the necessary features: community, posts, personal page management. It's all about implementing by CRUD.
  - Integrated with Camera AI, it can be used to scan objects and electronic devices and produce results of CO2 that those electronic devices emit. And let users know the amount of CO2 that the devices around them emit.
- Github: TylerTran4201/greendit_flutter.git

Tweet Clone

Personal project | 04/2023 - 05/2023

- Programming language: PHP
- Technology: Symfony, MariaDb, Tailwind Css
- Tools: Microsoft Visual Code, Docker
  - Develop a system where users can post posts, interact with posts such as liking and commenting. At the same time, you can follow the author.
  - The Admin side can manage information and perform basic CRUD operations in MariaDb.
- Github: TylerTran4201/clone-tweet-symfony